```cpp
1  ////
2  // Name: Andrew Norton
3  // Section: 132 Winter
4  // Program Name: ANString
5  //
6  // Description: Program takes a vector of a string-like class and fills it
7  // with words from a text-file. I then wrote the functions to sort the    ⮡
      strings alphabetically
8  // and then output to a file.
9  ////
10
11 #include <iostream>
12 #include <fstream>
13 #include <vector>
14 #include <iomanip>
15 #include "ANString.h"
16
17 using namespace std;
18 void BubbleSort(vector<ANString>& vect);
19 void output(vector<ANString>& vect);
20
21
22 int main() {
23
24     vector<ANString> words(100);        // calls default constructor 100   ⮡
          times
25     ifstream fin("infile3.txt");
26     ANString jumboString;
27     ANString empty;
28     int five = 0;
29     // File READ
30     if (fin.fail()) {
31         cout << "Couldn't open infile3.txt" << endl;
32         system("pause");
33         exit(1);
34     }
35
36     int wordCnt = 0;
37     for (; fin >> words[wordCnt] ; ++wordCnt) {        // empty loop
38
39      //    words[wordCnt
40        // cout << words[wordCnt].c_str() << '\t';       // for verifying    ⮡
            input
41     }
42     words.resize(wordCnt);
43     for (int i = 0; i < wordCnt; i++)
44     {
45         jumboString = jumboString + words[i];
46         five++;
```

```cpp
47            if (five == 5)
48            {
49                words.push_back(jumboString);
50                jumboString = empty;
51                five = 0;
52                wordCnt++;
53
54            }
55        }
56
57        if (five != 0)
58        {
59            wordCnt++;
60            words.push_back(jumboString);
61        }
62                //shrink vector to size used
63        BubbleSort(words);
64        output(words);
65        return 0;
66 }
67
68
69        // SORT
70        void BubbleSort(vector <ANString>& vect)
71        {
72
73            bool change = true;
74          while (change == true)
75            {
76                change = false;
77                for (int i = 0; i < vect.size() - 1; i++)
78                {
79                    // [debug statements] cout << (vect.at(i).c_str()) << " " << ⮐
                        (vect.at(i + 1).c_str()) << endl;
80                      if (vect[i] > vect[i+1])
81                      {
82                          change = true;
83                          ANString temp = vect[i];
84                          vect[i] = vect[i + 1];
85                          vect[i + 1] = temp;
86
87                          // cout << (vect.at(i).c_str()) << " " << (vect.at(i +   ⮐
                            1).c_str()) << endl;
88                      }
89                  }
90
91            }
92        }
93        void output(vector<ANString>& vect)
```

```cpp
 94      {
 95
 96          ofstream fout;
 97          fout.open("outfile.txt");
 98
 99          for (int i = 0; i < vect.size() - 1; i ++)
100          {    // += 6 needed so it doesn't repeat one of the 6 values on
                each line
101              fout << left << setw(13) << vect.at(i).c_str() << vect.at
                (i).length() << ":" << vect.at(i).capacity() << endl;
102
103          /*  for (int j = 0; j < 6; j++) // nested loop to only output 6
                words per line
104              {
105                  fout << left << setw(13) << vect.at(i+j).c_str();
106              }
107              fout << endl; */
108          }
109          fout << "Current number of classes: " << ANString::getCurrentCount
                () << endl;
110          fout << "Total classes created: " << ANString::getCreatedCount()
                << endl;
111          fout.close();
112          fout.clear();
113      }
114
115 /* Output
116 I          I          I          a          a          a
117 a          Or         an         an         as         in
118 in         or         to         to         to         to
119 to         my         be.        did        and        ask
120 ask        the        the        his        for        was
121 how        sit        Were       Such       They       life
122 seem       felt       much       time       what       they
123 they       each,      sits       man's      body?      busy.
124 began      began      brain      could      human      thing
125 steam      first      lower      rules      would      using,
126 seemed     engine     things     myself     myself     within
127 animal.    ruling,    Martian    compare    ironclad   machines,
128 amazingly  directing, impossible. mechanisms? intelligent
    intelligent
129 */
130
131
132
133
134
135
```